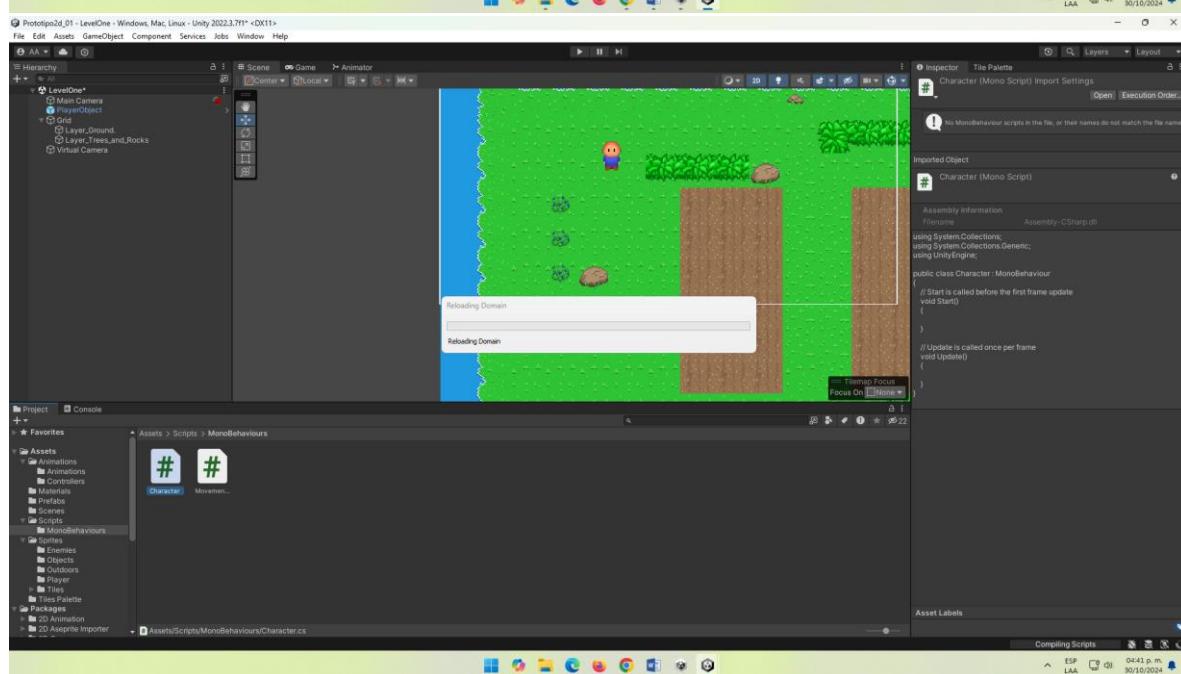
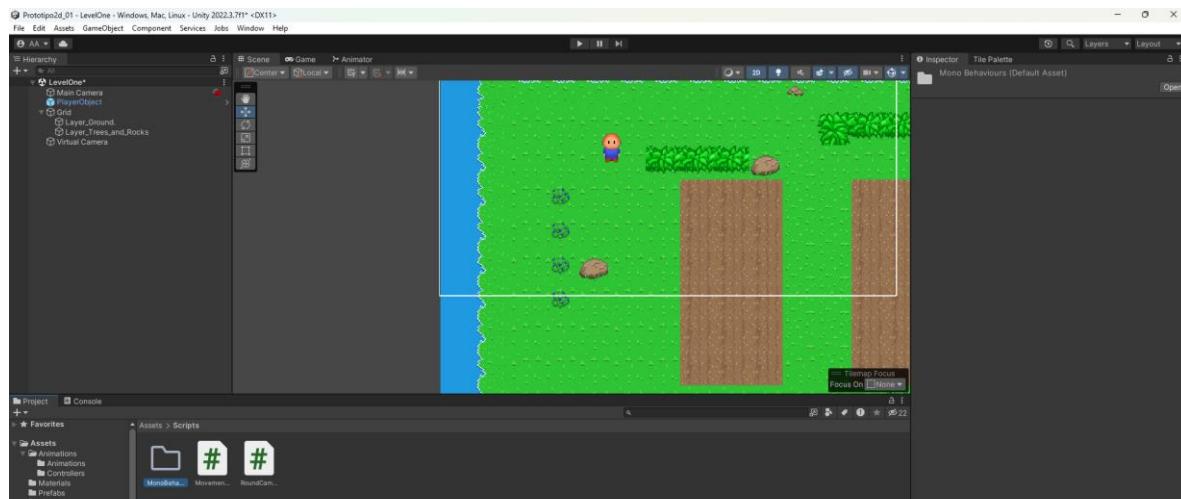


Parte 4



Vamos a construir una clase de personaje genérica a partir de la cual nuestra clase **Player** y **Enemy** heredarán. Esta clase **Character** contendrá funcionalidad y propiedades comunes a todos los tipos de personajes de nuestro juego.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

/*
Clase genérica para todo tipo de personaje en el Juego.
*/
public abstract class Character : MonoBehaviour
{
    public int hitPoints; //Puntos de vida actuales
    public int maxHitPoints; //Puntos de vida máximos
}
```

Modifico el archivo fuente

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

/*
Clase genérica para todo tipo de personaje en el Juego.
*/
public abstract class Character : MonoBehaviour
{
    public int hitPoints; //Puntos de vida actuales
    public int maxHitPoints; //Puntos de vida máximos
}

5
```

Clase Player

En nuestra carpeta **MonoBehaviours**, crea un nuevo script C# llamado **Player.cs**. Esta clase Player comenzará extremadamente simple, pero le agregaremos funcionalidad a medida que avancemos.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

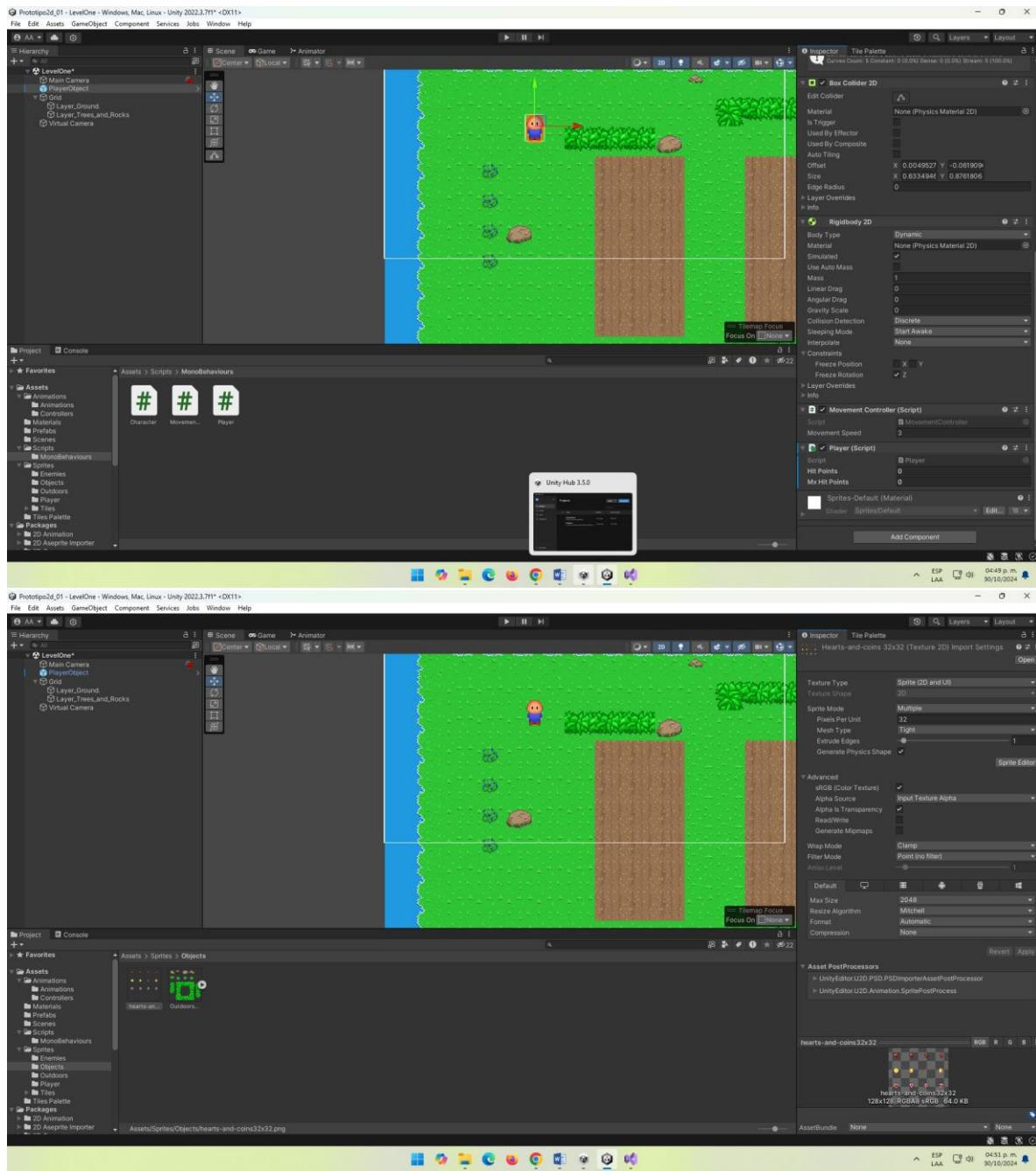
/*
Clase Player que hereda de Character
*/
public class Player : Character
{
    // Vacío por ahora
}
```

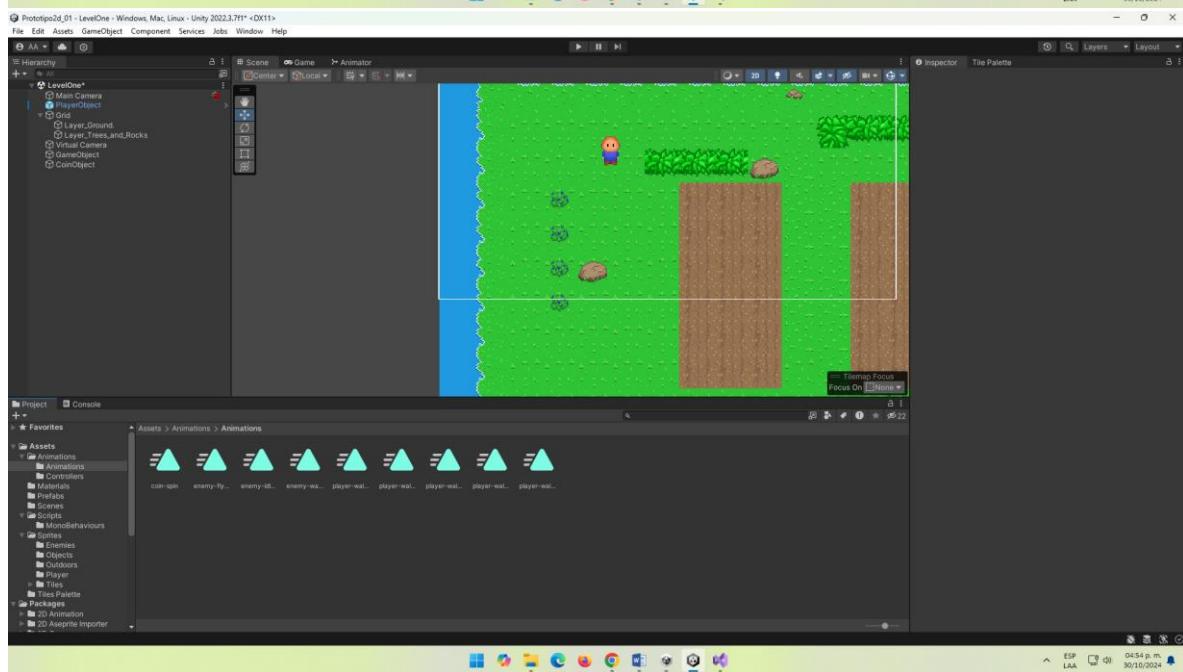
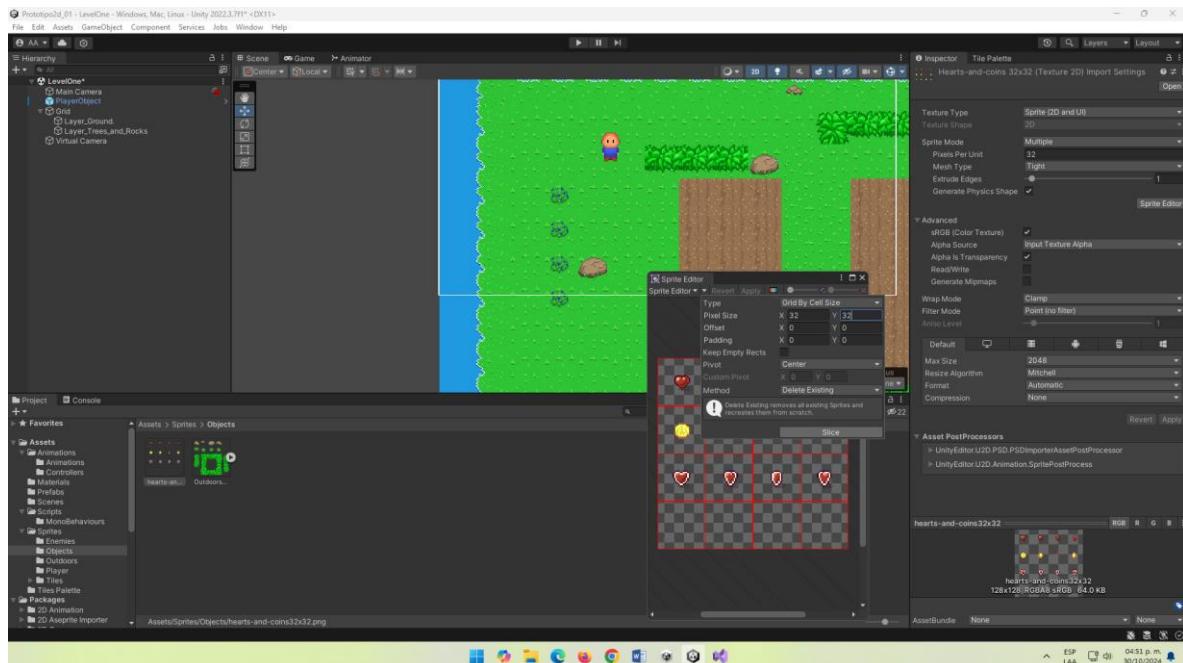
Todo lo que queremos hacer por ahora es heredar la clase **Character** para obtener la propiedad como **hitPoints**. Guarda el script.

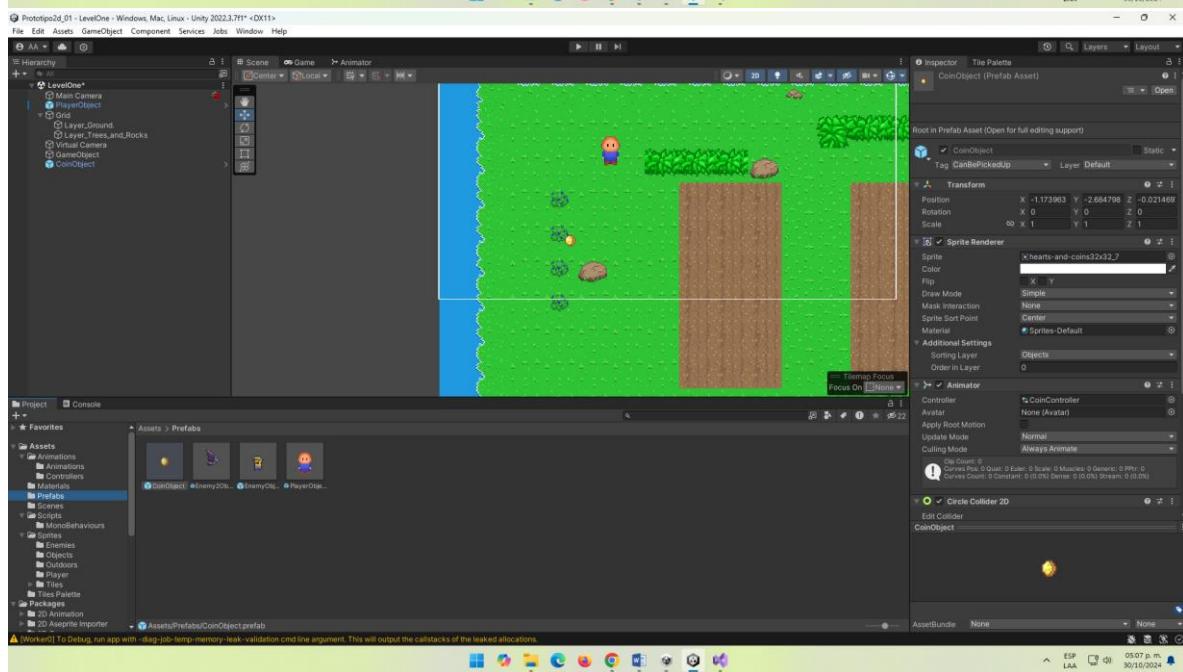
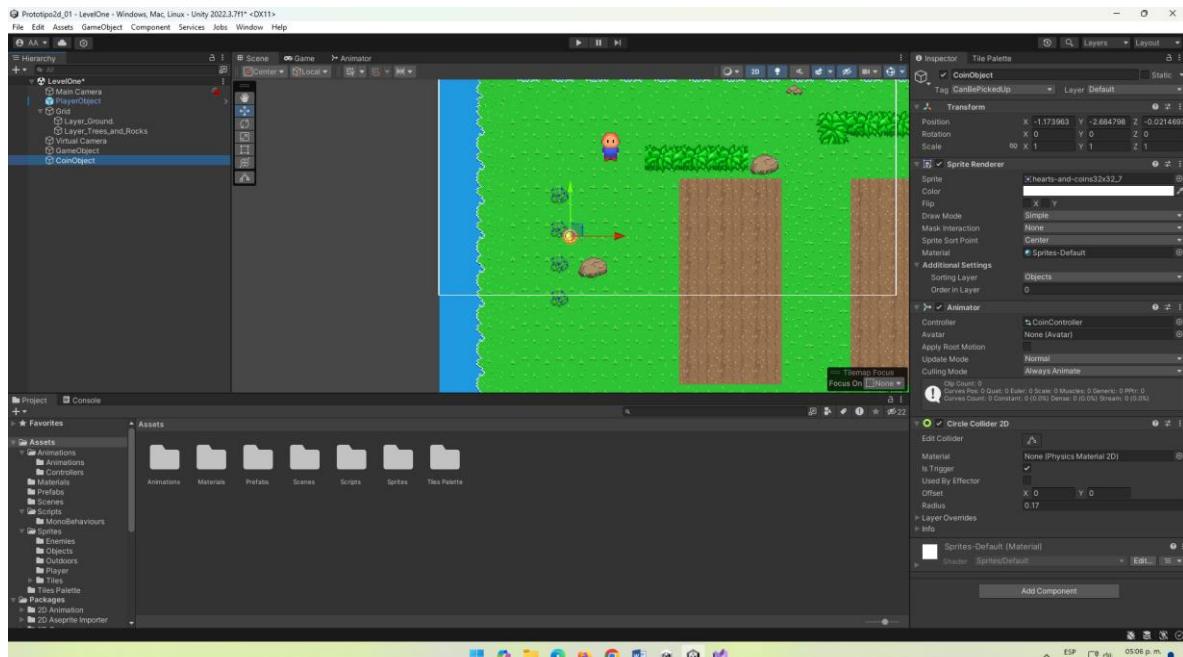
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

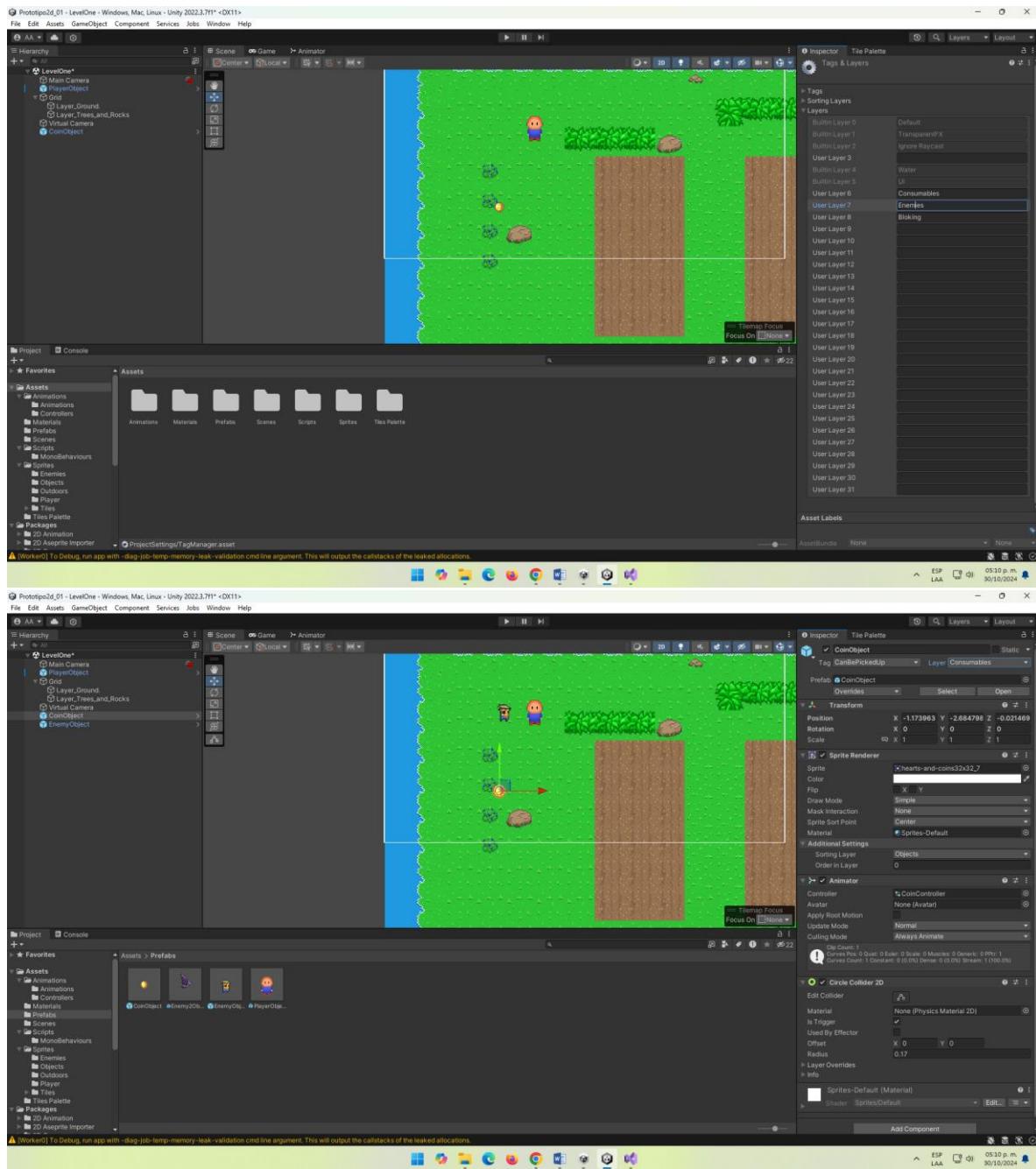
/*
Clase Player que hereda de Character
*/
public class Player : Character
{
    // Vacío por ahora
}

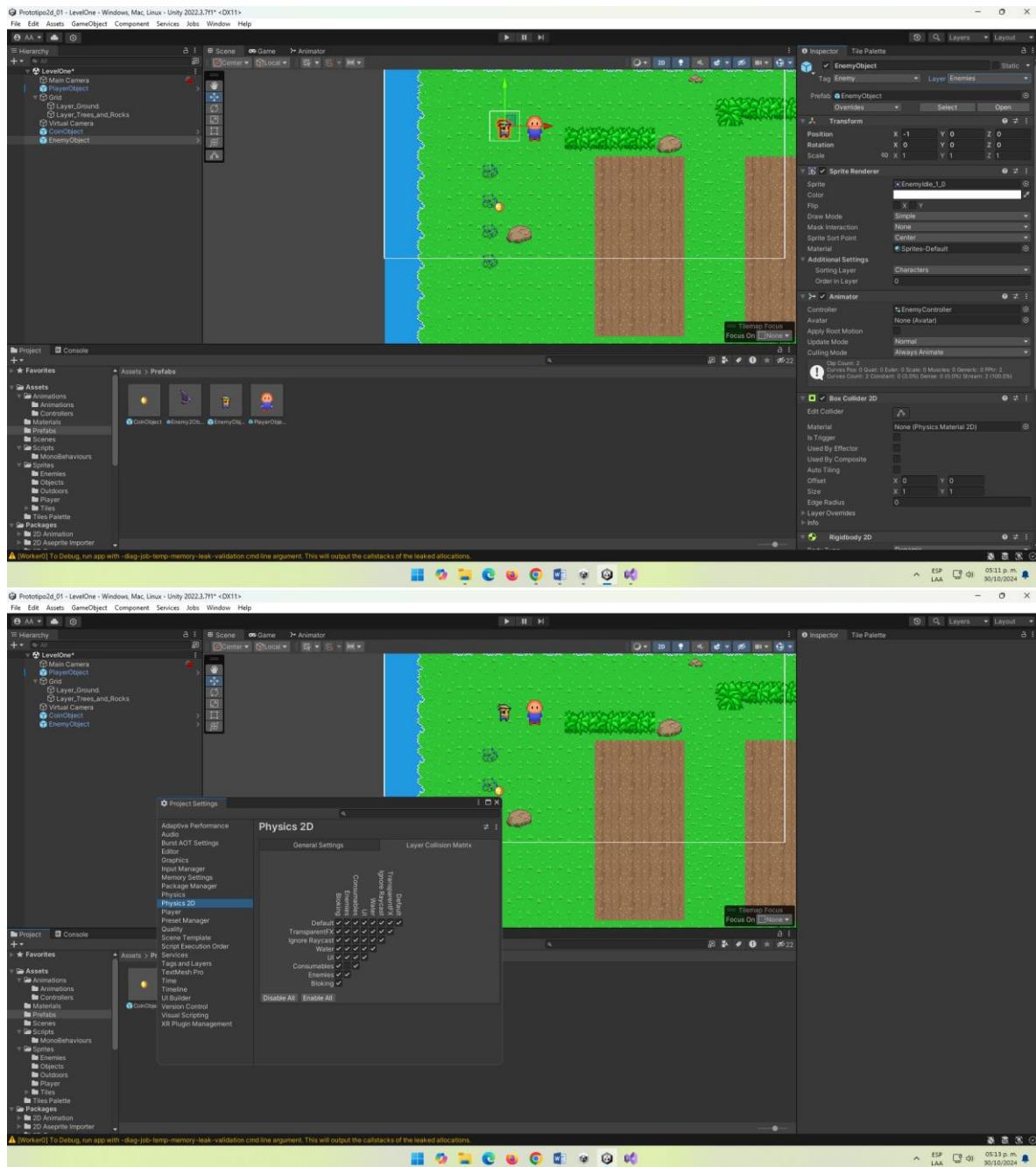
6
```

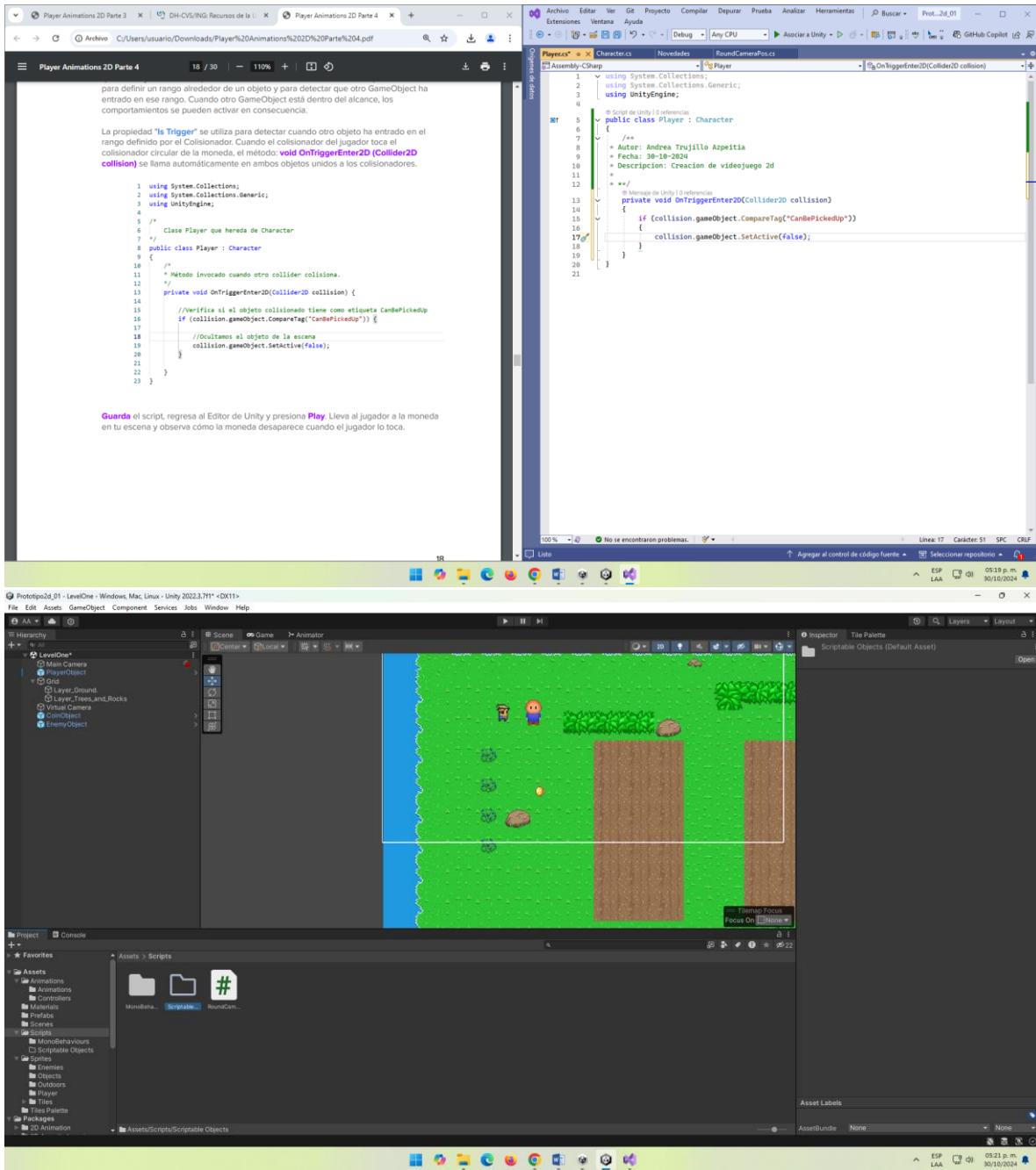


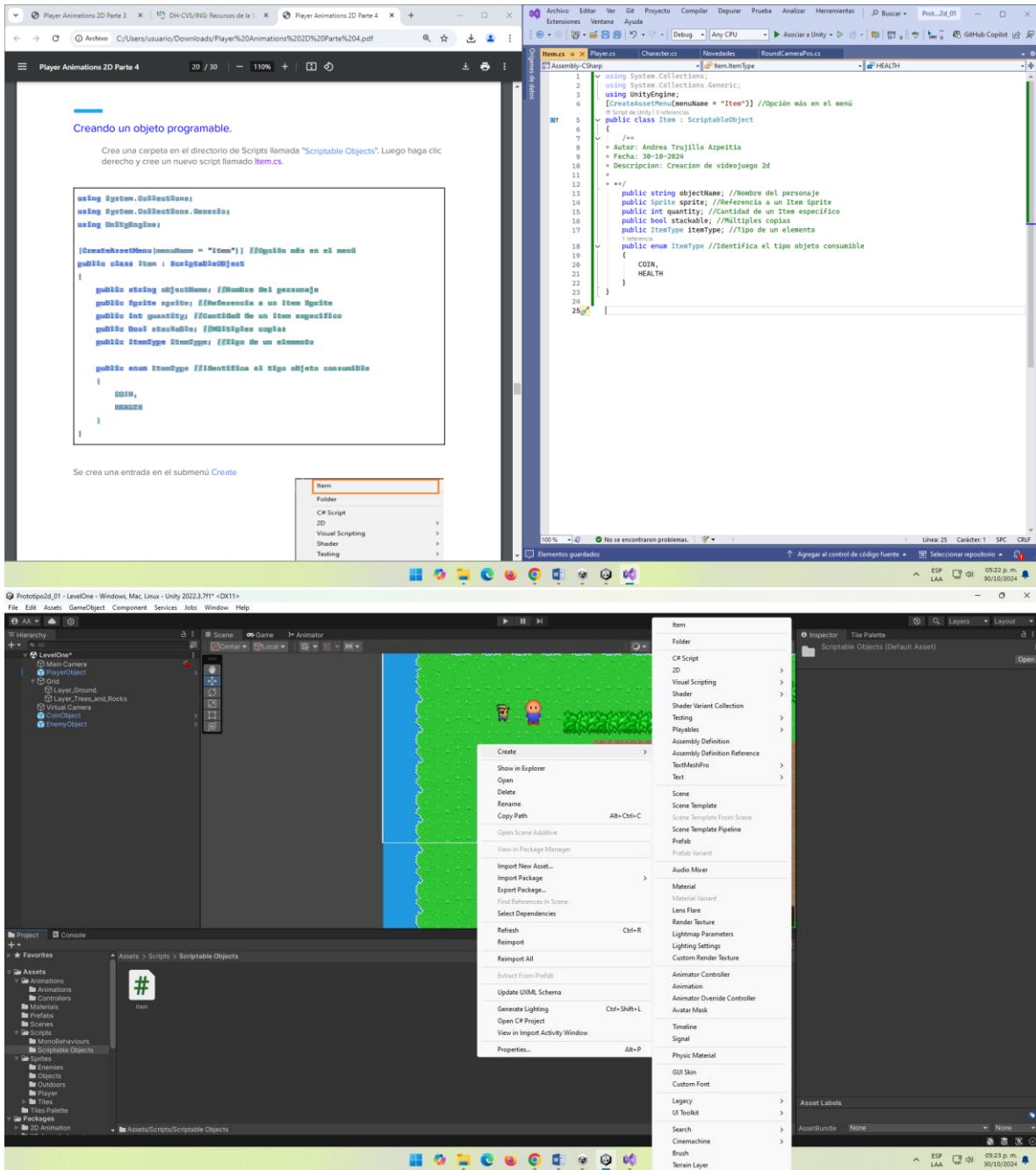












Player Animations 2D Parte 4

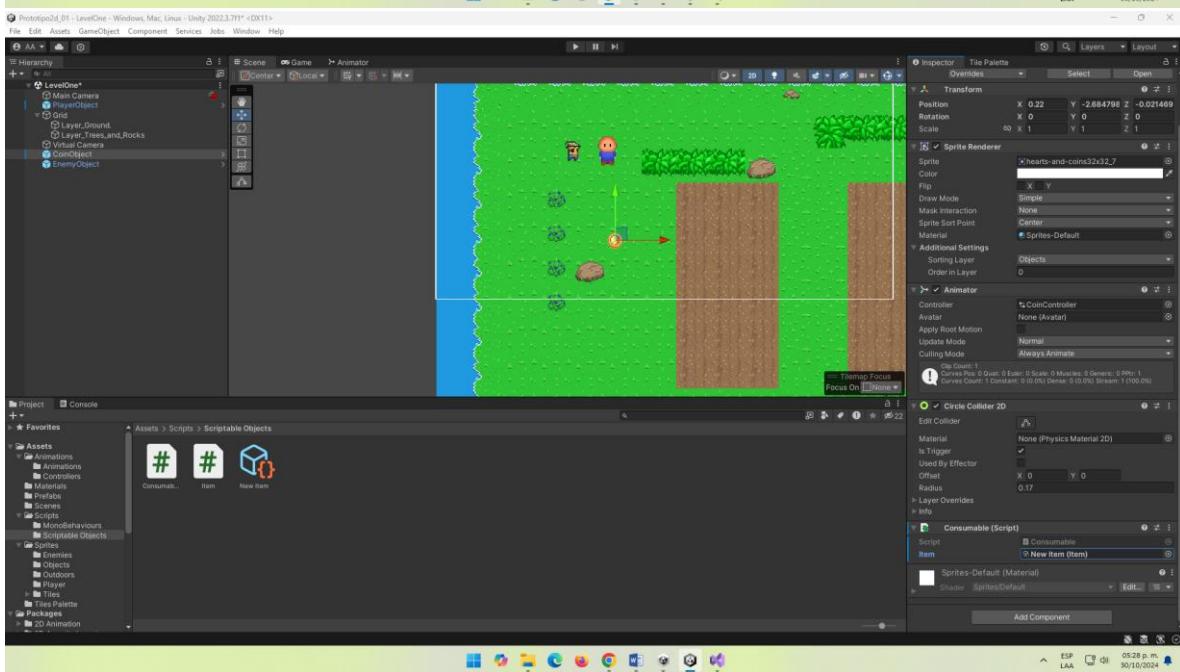
Consumable.cs

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

/*
 * Heredamos de MonoBehaviour lo podemos enlazar a un GameObject.
 */
public class Consumable : MonoBehaviour
{
    //Referencia a un objeto Scriptable o programable
    public Item item;
}

```



Player Animations 2D Parte 4 23 / 30 | - 110% +

Modifica el script **Player.cs**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

/*
    Clase Player que hereda de Character
*/
public class Player : Character
{
    /*
        * Método invocado cuando otro collider collisions.
    */
    private void OnTriggerEnter2D(Collider2D collision)
    {
        //Verifica si el objeto colisionado tiene como etiquetas CanBePickedUp
        if (collision.gameObject.CompareTag("CanBePickedUp"))
        {
            Item hitObject =
            collision.gameObject.GetComponent<Consumable>().item;

            if (hitObject != null)
            {
                print("Nombre: " + hitObject.objectName);
                collision.gameObject.SetActive(false);
            }
        }
    }
}

```

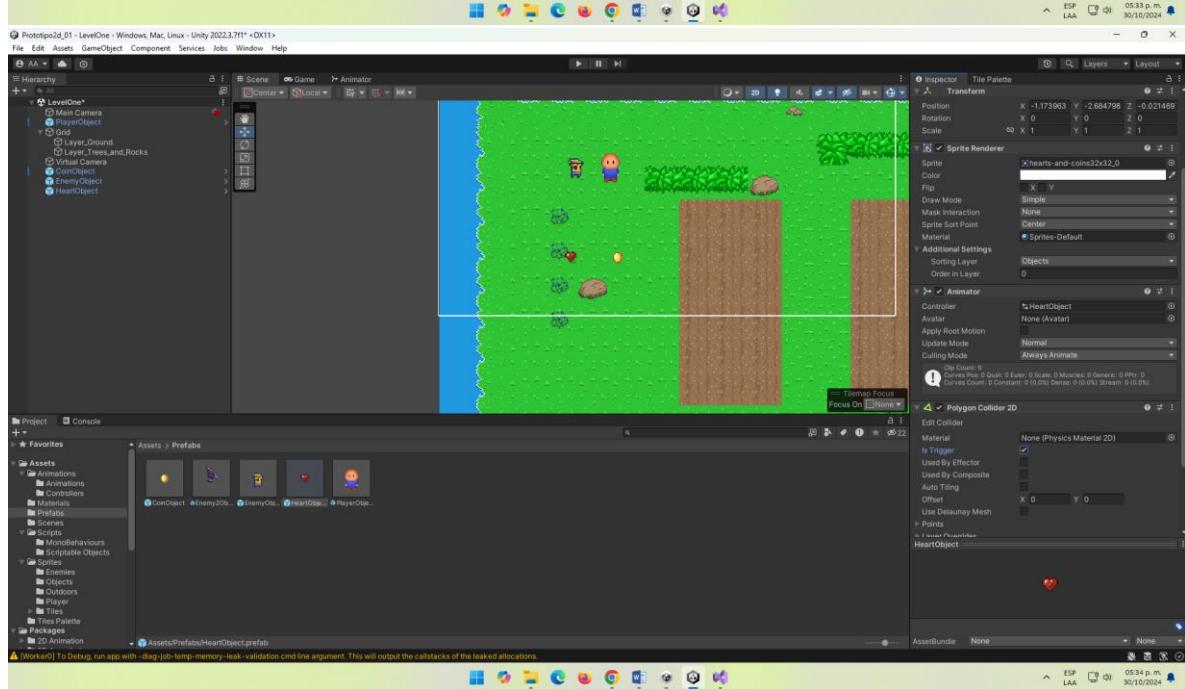
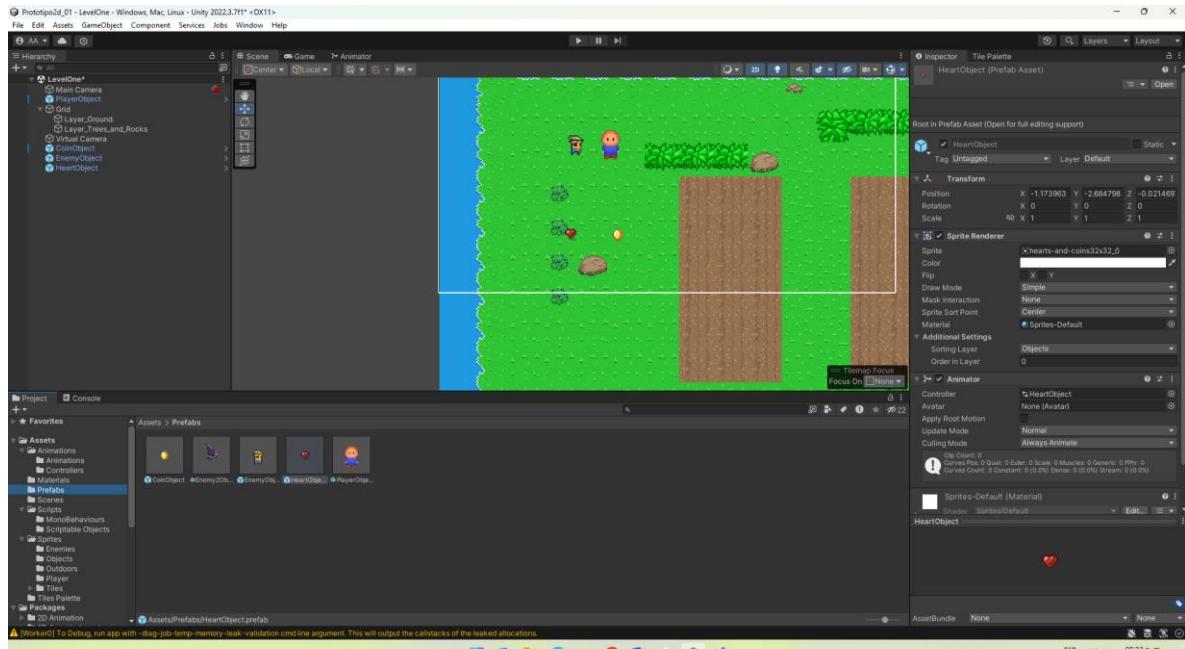
Assembly-Charts

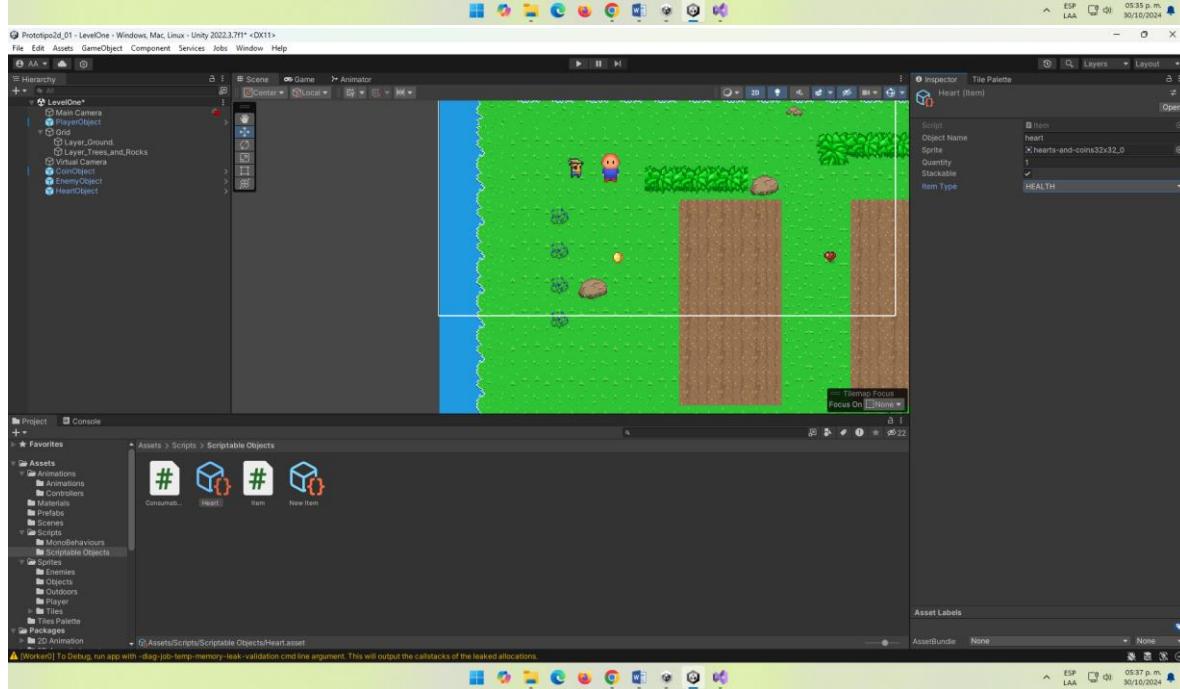
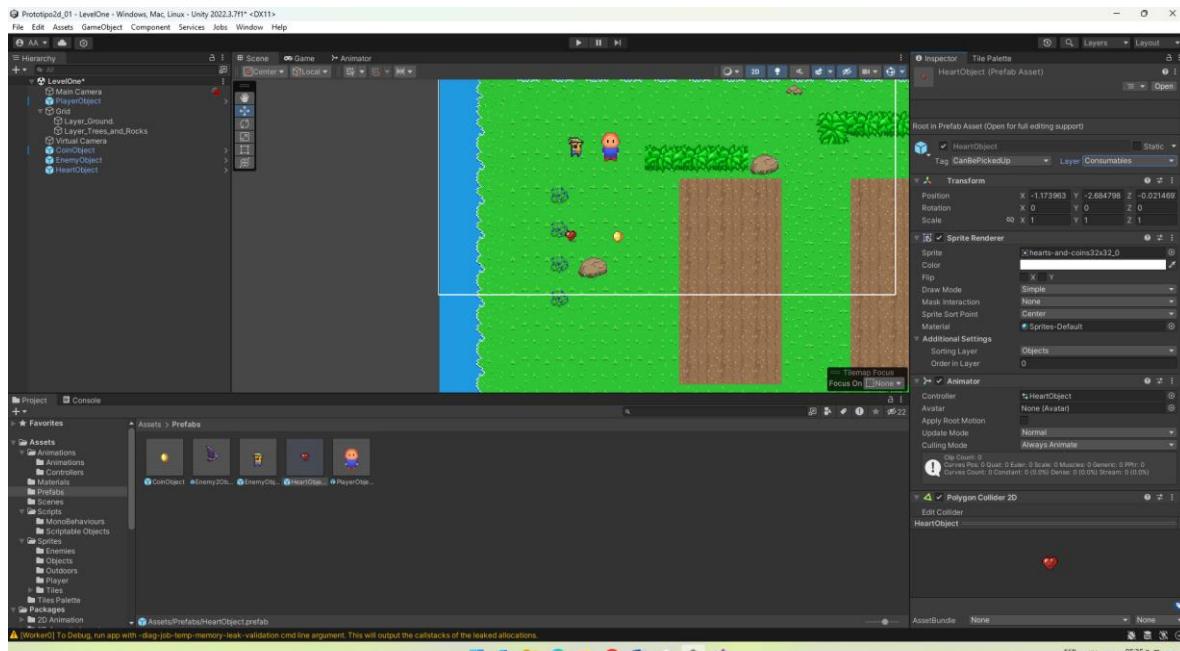
```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  // Script de Unity 1.0 referencias
6  public class Player : Character
7  {
8      /*
9          * Autor: Andrea Trujillo Arreitia
10         * Fecha: 30-10-2024
11         * Descripción: Creación de videojuego 2d
12     */
13
14     /*
15         * Método invocado cuando otro collider collisions.
16     */
17     // Mensaje de Unity 1.0 referencias
18     private void OnTriggerEnter2D(Collider2D collision)
19     {
20         //Verifica si el objeto colisionado tiene como etiqueta CanBePickedUp
21         if (collision.gameObject.CompareTag("CanBePickedUp"))
22         {
23             Item hitObject =
24             collision.gameObject.GetComponent<Consumable>().item;
25
26             if (hitObject != null)
27             {
28                 print("Nombre: " + hitObject.objectName);
29                 collision.gameObject.SetActive(false);
30             }
31         }
32     }
}

```

Prototipo2d_01 - LevelOne - Windows, Mac, Linux - Unity 2023.3.7f1 <DX11>





Player Animations 2D Parte 4

Guarda **Player.cs** vuelve al editor Unity.
Presiona **Play** y haz que el jugador corra hacia el objeto Prefab Heart. Debería ver el mensaje

```

if (hitObject != null)
{
    print("Nombre: " + hitObject.gameObject.name);

    switch (hitObject.itemType)
    {
        case Item.ItemType.COIN:
            break;
        case Item.ItemType.HEALTH:
            AdjustHitPoints(hitObject.quantity);
            break;
        default:
            break;
    }

    //Desactiva el objeto de la escena
    collision.gameObject.SetActive(false);
}

public void AdjustHitPoints(int amount)
{
    hitPoints += hitPoints + amount;
    print("Ajustando Puntos: " + amount + ", Nuevo Valor: " + hitPoints);
}

```

Linea 49 - Clickear 1 - SPC CRLF

